Extension of the Sparse Grid Quadrature Filter

Yang Cheng
Mississippi State University
Mississippi State, MS 39762
Email: cheng@ae.msstate.edu

Yang Tian
Harbin Institute of Technology
Harbin, Heilongjiang 150090, China
Email: tianyanghit@gmail.com

John L. Crassidis
University at Buffalo
Buffalo, NY 14260
Email: johnc@buffalo.edu

***Abstract*—The sparse grid quadrature filter is a point-based Gaussian filter in which expectations of nonlinear functions of Gaussian random vectors are computed using the sparse grid quadrature. The sparse grid quadrature can be considered a generalization of the Unscented Transform in that the Unscented Transform is equivalent to the level-2 sparse grid quadrature. A novel extension of the sparse grid quadrature filter is presented that directly transforms the points in time update and measurement update to eliminate repeated covariance decomposition based point generation and to relax the Gaussian assumption inherent in the sparse grid quadrature filter as well as the sigma-point filters. A tracking example is presented to demonstrate the performance of the novel filter.**

I. Introduction

Gaussian filters [1], [2] are approximate Bayesian filters derived under the assumption that all probability density functions (pdfs) in the system are Gaussian. Because a Gaussian pdf is fully characterized by the mean and covariance, the recursion of the posterior pdf is replaced with the recursion of the mean and covariance, where the multi-dimensional Gaussian weighted integrals are computed using quadrature or cubature methods [3]. Most Gaussian filters share the same structure and differ only in how the Gaussian weighted integrals are numerically computed. The quadrature-based Gaussian filters, including the sigma-point filters, are also called point-based Gaussian filters.

Many quadrature or cubature rules can be used to approximate the Gaussian weighted integrals, including but not limited to the unscented transformation [4], [5], [3], the Gauss-Hermite quadrature[1], the spherical-radial cubature [6], [7], [8], and the sparse-grid quadrature[9], [10]. The corresponding quadrature- or point-based Gaussian filters are the unscented Kalman filters[4], [5], the Gauss-Hermite quadrature filter[1], the cubature Kalman filters[6], [8], and the sparse-grid quadrature filters[9], [10], respectively. The cubature Kalman filter of [6] is a special case of the unscented Kalman filter of [4], which in turn is a special case of the sparse grid quadrature filters [9], [10] or the arbitrary-degree cubature Kalman filters [8].

Roughly speaking, the computational complexity of the point-based Gaussian filter is proportional to the number of points used to compute the Gaussian weighted integrals. The computational complexity of the Gauss-Hermite quadrature filter increases exponentially with the dimension of the system [1]. The sparse grid quadrature filters [9], [10] as well as the high-degree cubature Kalman filters of [8] are closely related to the Gauss-Hermite quadrature filter, but has polynomial computational complexity (linear complexity included). The sparse grid quadrature filters and cubature Kalman filters with linear computational complexity are equivalent to many sigma-point filters.

For linear Gaussian systems, where the posterior pdf is exactly Gaussian, all the quadrature-based Gaussian filters are equivalent to the optimal linear Kalman filter and are thus equally accurate. When applied to nonlinear systems, the accuracy of the point-based Gaussian filters is to a large extent determined by the quadrature or cubature accuracy. The Gauss-Hermite quadrature filter can compute the Gaussian weighted integrals to highest accuracy (at highest computational cost) and is the most accurate Gaussian filter. The sparse grid quadrature filter has best balance between accuracy and computational complexity.

The accuracy of the Gaussian filters depends on the validity of the Gaussian assumption as well. In the presence of strong nonlinearity, the true pdf of the state may be noticeably non-Gaussian. In these cases, the partial information about the underlying non-Gaussian distribution contained in the quadrature points should be retained and exploited [11]. A scheme that directly updates the points in the measurement update was proposed in [11]. This paper presents a full extension of the sparse grid quadrature filter, where the Gaussian assumption and the re-regeneration of the quadrature points according to the Gaussian assumption are removed. In both the time update and measurement update of this extended sparse grid quadrature filter, the quadrature points are directly updated to match the mean and covariance, which are computed separately using the formulae of the sparse grid quadrature filter. Like the quadrature-based Gaussian filters, the extended sparse grid quadrature filter reduces to the optimal linear Kalman filter when the system is linear Gaussian.

The remainder of the paper is organized as follows. Section II reviews the quadrature-based Gaussian filters including the sparse grid quadrature filter. The extended sparse grid quadrature filter is presented in Section III. An illustrative example of radar tracking of a vertically falling object is given in Section IV, followed by the Conclusions.

II. Quadrature-Based Gaussian Filtering

A. Gaussian Filtering

The Gaussian filter simplifies the recursion of the posterior pdf by *approximating* the state pdfs $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ and $p(\mathbf{x}_{k+1}|\mathbf{z}_{1:k})$, where $\mathbf{z}_{1:k} = \{\mathbf{z}_0, \mathbf{z}_1, \ldots, \mathbf{z}_k\}$, the state transition density $f_{k-1}(\mathbf{x}_k|\mathbf{x}_{k-1})$, and the likelihood function $g_k(\mathbf{z}_k|\mathbf{x}_k)$ by Gaussian pdfs. It is well known that a Gaussian pdf is parameterized by the mean $\mathbf{m}$ and covariance P [1]

$$\mathcal{N}(\mathbf{x};\mathbf{m},P)=\frac{\exp\left[-\frac{1}{2}\left(\mathbf{x}-\mathbf{m}\right)^T P^{-1}\left(\mathbf{x}-\mathbf{m}\right)\right]}{\left[\det(2\pi P)\right]^{1/2}} \tag{1}$$

where $\det$ denotes the matrix determinant and the superscript T denotes the matrix or vector transpose. In Gaussian filters, the recursion of the state pdf is completely replaced by the recursion of the mean and covariance of the (Gaussian) pdf.

The Gaussian filter is usually applied to the following discrete-time system:

$$\mathbf{x}_k=\mathbf{f}_{k-1}\left(\mathbf{x}_{k-1}\right)+\mathbf{w}_k \tag{2a}$$

$$\mathbf{z}_k=\mathbf{g}_k\left(\mathbf{x}_k\right)+\boldsymbol{\nu}_k \tag{2b}$$

where the dimensions of $\mathbf{x}_k$ and $\mathbf{z}_k$ are n and m, respectively, the additive process noise $\mathbf{w}_k\sim\mathcal{N}(\mathbf{w};\mathbf{0},Q_k)$, which means that $\mathbf{w}_k$ is sampled from the Gaussian probability density with mean zero and covariance Q_k, and the additive Gaussian measurement noise $\boldsymbol{\nu}_k\sim\mathcal{N}(\boldsymbol{\nu};\mathbf{0},R_k)$, which means that $\boldsymbol{\nu}_k$ is sampled from the Gaussian probability density with mean zero and covariance R_k.

A cycle of the Gaussian filter consists of a time update step and a measurement update step. For the time interval $[k-1,k]$, assume

$$p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})\approx\mathcal{N}(\mathbf{x}_{k-1};\hat{\mathbf{x}}_{k-1}^{+},P_{k-1}^{+}) \tag{3a}$$

$$p(\mathbf{x}_{k}|\mathbf{z}_{1:k-1})\approx\mathcal{N}(\mathbf{x}_{k};\hat{\mathbf{x}}_{k}^{-},P_{k}^{-}) \tag{3b}$$

$$p(\mathbf{x}_{k}|\mathbf{z}_{1:k})\approx\mathcal{N}(\mathbf{x}_{k};\hat{\mathbf{x}}_{k}^{+},P_{k}^{+}) \tag{3c}$$

The time update step computes $\hat{\mathbf{x}}_k^-$ and P_k^- from $\hat{\mathbf{x}}_{k-1}^+$ and P_{k-1}^+. The measurement update step computes $\hat{\mathbf{x}}_k^+$ and P_k^+ from $\hat{\mathbf{x}}_k^-$ and P_k^-.

The time update is given by [1]

$$\hat{\mathbf{x}}_k^-=\int_{\mathbb{R}^n}\mathbf{f}_{k-1}(\boldsymbol{\xi})\mathcal{N}(\boldsymbol{\xi};\hat{\mathbf{x}}_{k-1}^{+},P_{k-1}^{+})\,d\boldsymbol{\xi} \tag{4a}$$

$$P_k^-=\int_{\mathbb{R}^n}\tilde{\boldsymbol{\xi}}\tilde{\boldsymbol{\xi}}^T\mathcal{N}(\boldsymbol{\xi};\hat{\mathbf{x}}_{k-1}^{+},P_{k-1}^{+})\,d\boldsymbol{\xi}+Q_k \tag{4b}$$

with $\tilde{\boldsymbol{\xi}}=\mathbf{f}_{k-1}(\boldsymbol{\xi})-\hat{\mathbf{x}}_k^-$.

The measurement update is based on linear minimum mean square error estimation, which updates the mean and covariance based on correlation matrices. It is given by [1]

$$\hat{\mathbf{x}}_k^+=\hat{\mathbf{x}}_k^-+K_k\left(\mathbf{z}_k-\hat{\mathbf{z}}_k^-\right) \tag{5a}$$

$$P_k^+=P_k^--K_k(P_k^{xz})^T \tag{5b}$$

with

$$\hat{\mathbf{z}}_k^-=\int_{\mathbb{R}^n}\mathbf{g}_k(\boldsymbol{\xi})\mathcal{N}(\boldsymbol{\xi};\hat{\mathbf{x}}_k^-,P_k^-)d\boldsymbol{\xi} \tag{6a}$$

$$P_k^{xz}=\int_{\mathbb{R}^n}(\boldsymbol{\xi}-\hat{\mathbf{x}}_k^-)(\mathbf{g}_k(\boldsymbol{\xi})-\hat{\mathbf{z}}_k^-)^T\mathcal{N}(\boldsymbol{\xi};\hat{\mathbf{x}}_k^-,P_k^-)d\boldsymbol{\xi} \tag{6b}$$

$$P_k^{zz}=\int_{\mathbb{R}^n}\left(\mathbf{z}_k-\mathbf{g}_k(\boldsymbol{\xi})\right)\left(\mathbf{z}_k-\mathbf{g}_k(\boldsymbol{\xi})\right)^T\mathcal{N}(\boldsymbol{\xi};\hat{\mathbf{x}}_k^-,P_k^-)d\boldsymbol{\xi} \tag{6c}$$

$$K_k=P_k^{xz}(P_k^{zz}+R_k)^{-1} \tag{6d}$$

where the superscript -1 denotes the matrix inverse.

The Gaussian filters can also be developed for systems with non-additive process noise and/or measurement noise [5].

B. Gaussian Quadratures

All n-dimensional integrals in the time update and measurement update of the Gaussian filter in Eqs. (4) and (6) are of the form

$$\int_{\mathbb{R}^n}\boldsymbol{\mathcal{G}}(\boldsymbol{\xi})\mathcal{N}(\boldsymbol{\xi};\mathbf{m},P)\,d\boldsymbol{\xi}$$

where $\boldsymbol{\mathcal{G}}(\cdot)$ is a general nonlinear function. The quadrature-based Gaussian filters approximate the Gaussian weighted integrals using numerical integration methods. The integral can be rewritten as

$$\int_{\mathbb{R}^n}\boldsymbol{\mathcal{G}}(\boldsymbol{\xi})\mathcal{N}(\boldsymbol{\xi};\mathbf{m},P)\,d\boldsymbol{\xi}=\int_{\mathbb{R}^n}\boldsymbol{\mathcal{G}}(S\boldsymbol{\xi}+\mathbf{m})\mathcal{N}(\boldsymbol{\xi};\mathbf{0},I_{n\times n})\,d\boldsymbol{\xi} \tag{7}$$

where the matrix S satisfies

$$P=SS^T \tag{8}$$

Note that S is not unique. If the quadrature approximation for $\mathcal{N}(\boldsymbol{\xi};\mathbf{0},I_{n\times n})$ is given by

$$\int_{\mathbb{R}^n}\boldsymbol{\mathcal{G}}(\boldsymbol{\xi})\mathcal{N}(\boldsymbol{\xi};\mathbf{0},I_{n\times n})\,d\boldsymbol{\xi}\approx\sum_{i=1}^{N}w^{(i)}\boldsymbol{\mathcal{G}}(\boldsymbol{\chi}^{(i)}) \tag{9}$$

where $\boldsymbol{\chi}^{(i)}$ and $w^{(i)}$, $i=1,\ldots,N$, are the quadrature points and their associated weights for $\mathcal{N}(\mathbf{x};\mathbf{0},I_{n\times n})$, we can use the same quadrature points $\boldsymbol{\chi}^{(i)}$ and weights $w^{(i)}$ to approximate $\int_{\mathbb{R}^n}\boldsymbol{\mathcal{G}}(\boldsymbol{\xi})\mathcal{N}(\boldsymbol{\xi};\mathbf{m},P)\,d\boldsymbol{\xi}$, given by

$$\int_{\mathbb{R}^n}\boldsymbol{\mathcal{G}}(\boldsymbol{\xi})\mathcal{N}(\boldsymbol{\xi};\mathbf{m},P)\,d\boldsymbol{\xi}\approx\sum_{i=1}^{N}w^{(i)}\boldsymbol{\mathcal{G}}(S\boldsymbol{\chi}^{(i)}+\mathbf{m}) \tag{10}$$

The above equation indicates that the quadrature points and weights for $\mathcal{N}(\boldsymbol{\xi};\mathbf{m},P)$ are given by $S\boldsymbol{\chi}^{(i)}+\mathbf{m}$ and $w^{(i)}$, respectively. The quadrature points for the Gaussian random vector are usually symmetric.

The main requirement for the quadrature points $\boldsymbol{\chi}^{(i)}$ and weights $w^{(i)}$ is that they match some of the moments of a Gaussian random vector $\mathbf{x}$ exactly, which is equivalent to computing the integrals of the monomial $\mathbf{x}^d$ exactly. Here $\mathbf{x}^d$ is defined as

$$\mathbf{x}^d=\prod_{j=1}^{n}x_j^{\alpha_j} \tag{11}$$

with $\sum_{j=1}^{n}\alpha_j=d$, and d is the total degree of the monomial $\mathbf{x}^d$. The integral of the monomial with degree d is given by $\int_{\mathbb{R}^n}\boldsymbol{\xi}^d\mathcal{N}(\boldsymbol{\xi};\mathbf{m},P)\,d\boldsymbol{\xi}$.

Note that given the moment matching requirement, there exist an infinite number of point sets that satisfy it.

C. Sparse Grid Quadrature

The sparse grid quadrature [9], [10] is an accurate and efficient method for computing the expectations of nonlinear functions with respect to a known pdf. In the Gaussian filter, the pdf is assumed to be Gaussian.

The sparse grid points $\boldsymbol{\mathcal{X}}^{(l)}$ for $\mathcal{N}\left(\mathbf{x};\mathbf{m},P\right)$ are related to the sparse grid points $\boldsymbol{\chi}^{(l)}$ for $\mathcal{N}\left(\mathbf{x};\mathbf{0},I\right)$ by

$$\boldsymbol{\mathcal{X}}^{(l)}=\sqrt{P}\boldsymbol{\chi}^{(l)}+\mathbf{m} \tag{12}$$

where $\sqrt{P}\sqrt{P}^T = P$. The matrix $\sqrt{P}$ is not unique.

The n-dimensional sparse grid quadrature is built from univariate or one-dimensional quadratures.

A general univariate quadrature rule $\mathcal{I}(g)$ is given by

$$\mathcal{I}(g) = \sum_{i=1}^{M} \varpi^{(i)} g(x^{(i)}) \approx \int g(x) w(x) dx \tag{13}$$

where $w(x)$ is a weighting function, $x^{(i)}$ and $\varpi^{(i)}$ are selected to achieve specified accuracy, and $\varpi^{(i)} > 0$, $\sum_{i=1}^{m} \varpi^{(i)} = 1$. The expectation with respect to the standard Gaussian density $\mathcal{N}(x; 0, 1)$ is related to Gauss-Hermite quadrature, with $w(x) = \exp(-x^2)$, $x \in (-\infty, \infty)$, in (13), by

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} g(x) \exp(-\frac{x^2}{2})\, dx = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} g(\sqrt{2}x) \exp(-x^2)\, dx \tag{14}$$

The left-hand side of the equation is the expectation with respect to the standard Gaussian density and the right-hand side of the equation is the integration to be approximated by Gauss-Hermite quadrature. A similar relationship exists between the expectation with respect to $\mathcal{N}(\mathbf{x}; \mathbf{0}, I)$ and the n-dimensional Gauss-Hermite quadrature.

A univariate quadrature $\mathcal{I}$ is said to have accuracy level m if it is exact for all polynomials $g(x)$ up to degree $2m-1$. A level-m univariate quadrature will be denoted by $\mathcal{I}_m$.

An n-dimensional sparse grid quadrature for $\mathbf{x} = [x_1, \ldots, x_n]$ has accuracy level L if it is exact for all polynomials $x_1^{j_1} x_2^{j_2} \cdots x_n^{j_n}$ with $\sum_{k=1}^{n} j_k \le 2L-1$. Such a sparse grid quadrature is denoted by $\mathcal{I}_{n,L}^{S}$.

An n-dimensional Gauss-Hermite quadrature with accuracy level L is generated from univariate Gauss-Hermite quadratures $\mathcal{I}_l$ with accuracy level l, $1 \le l \le L$.

The Smolyak quadrature rule is used to construct the n-dimensional sparse grid quadrature:

$$\mathcal{I}_{n,L}^{S} = \sum_{\Xi \in \mathbf{\Upsilon}_{n,L}} (\Delta_{j_1} \otimes \Delta_{j_2} \otimes \cdots \otimes \Delta_{j_n}) \tag{15}$$

with $\otimes$ the tensor product, $\Delta_{j_k} = \mathcal{I}_{j_k} - \mathcal{I}_{j_{k-1}}$, $\mathcal{I}_0 = 0$, $\Xi = (j_1, \cdots, j_n)$, and

$$\mathbf{\Upsilon}_{n,L} = \left\{ \Xi = (j_1, \cdots, j_n) : j_k \ge 1, \sum_{k=1}^{n} (j_k - 1) \le (L-1) \right\} \tag{16}$$

By introducing an auxiliary nonnegative integer q and defining

$$\mathbf{N}_q^n = \begin{cases} \{\Xi \subset \mathbf{N}^n : \sum_{k=1}^{n} j_k = n+q\} & q \ge 0 \\ \varnothing & q < 0 \end{cases}, \tag{17}$$

Equation (15) can be rewritten as

$$\mathcal{I}_{n,L}^{S} = \sum_{q=L-n}^{L-1} (-1)^{L-1-q} \binom{n-1}{L-1-q} \sum_{\Xi \in \mathbf{N}_q^n} (\mathcal{I}_{j_1} \otimes \mathcal{I}_{j_2} \otimes \cdots \mathcal{I}_{j_n}) \tag{18}$$

The sparse grid points and weights for an n-dimensional sparse grid quadrature with an accuracy level of L can be generated based on the above equation using sophisticated but standard algorithms. By the definition of accuracy level, the level-L sparse grid points and weights are capable of capturing the up to $(2L-1)$-th moments of $\mathcal{N}(\mathbf{x}; \mathbf{0}, I)$ or $\mathcal{N}(\mathbf{x}; \mathbf{m}, P)$.

D. Quadrature-Based Gaussian Filtering

The quadrature-based Gaussian filters differ only in the quadrature used. When the sparse grid quadrature is used, the Gaussian filter is the sparse grid quadrature filter. A cycle of the quadrature-based Gaussian filter is given as follows:

1) Time Update
   a) Given $\hat{\mathbf{x}}_{k-1}^{+}$ and P_{k-1}^{+}, generate the quadrature points $\mathbf{X}^{(i)}$ and weights $w^{(i)}$ for $\mathcal{N}(\mathbf{x}_{k-1}; \hat{\mathbf{x}}_{k-1}^{+}, P_{k-1}^{+})$
   b) Compute $\hat{\mathbf{x}}_k^{-}$ and P_k^{-}

$$\hat{\mathbf{x}}_k^{-} = \sum_{i=1}^{N} w^{(i)} \mathbf{f}_{k-1}(\mathbf{X}^{(i)}) \tag{19a}$$

$$P_k^{-} = \sum_{i=1}^{N} w^{(i)} \tilde{\mathbf{X}}^{(i)} (\tilde{\mathbf{X}}^{(i)})^T + Q_k \tag{19b}$$

   where $\tilde{\mathbf{X}}^{(i)} = \mathbf{f}_{k-1}(\mathbf{X}^{(i)}) - \hat{\mathbf{x}}_k^{-}$
   c) Re-generate the quadrature points $\mathbf{X}^{(i)}$ and weights $w^{(i)}$ for $\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_k^{-}, P_k^{-})$ so that the quadrature points account for the effect of the process noise
2) Measurement Update
   a) Compute $\hat{\mathbf{z}}_k^{-}$, P_k^{xz}, and P_k^{zz}

$$\mathbf{Z}^{(i)} = \mathbf{g}_k(\mathbf{X}^{(i)}) \tag{20a}$$

$$\hat{\mathbf{z}}_k^{-} = \sum_{i=1}^{N} w^{(i)} \mathbf{Z}^{(i)} \tag{20b}$$

$$P_k^{xz} = \sum_{i=1}^{N} w^{(i)} (\mathbf{X}^{(i)} - \hat{\mathbf{x}}_k^{-})(\mathbf{Z}^{(i)} - \hat{\mathbf{z}}_k^{-})^T \tag{20c}$$

$$P_k^{zz} = \sum_{i=1}^{N} w^{(i)} (\mathbf{Z}^{(i)} - \hat{\mathbf{z}}_k^{-})(\mathbf{Z}^{(i)} - \hat{\mathbf{z}}_k^{-})^T + R_k \tag{20d}$$

   b) Compute the Kalman gain K_k

$$K_k = P_k^{xz} (P_k^{zz} + R_k)^{-1} \tag{21}$$

   c) Compute $\hat{\mathbf{x}}_k^{+}$ and P_k^{+}

$$\hat{\mathbf{x}}_k^{+} = \hat{\mathbf{x}}_k^{-} + K_k \left(\mathbf{z}_k - \hat{\mathbf{z}}_k^{-}\right) \tag{22a}$$

$$P_k^{+} = P_k^{-} - K_k (P_k^{xz})^T \tag{22b}$$

The Gaussian filters recursively compute $\hat{\mathbf{x}}_k^{-}$, P_k^{-}, $\hat{\mathbf{x}}_k^{+}$, and P_k^{+}. The quadrature points in the filters have a secondary role and are re-generated from $\hat{\mathbf{x}}_k^{-}$ and P_k^{-} and from $\hat{\mathbf{x}}_k^{+}$ and P_k^{+} in every filter cycle.

III. Extended Sparse Grid Quadrature Filter

The basic idea of the extended sparse grid quadrature filter is to update the quadrature points directly under the constraint that the mean and covariance of the updated quadrature points match $\hat{\mathbf{x}}_k^-$ and P_k^- in time update and $\hat{\mathbf{x}}_k^+$ and P_k^+ in measurement update.

1) Initialization: Generate the quadrature points $\mathbf{X}_0^{(i)+}$ and weights $w^{(i)}$ for $\mathcal{N}(\mathbf{x}_0; \hat{\mathbf{x}}_0^+, P_0^+)$
2) Time Update
   a) Propagate the quadrature points
$$\mathbf{X}_k^{(i)-} = \mathbf{f}_{k-1}(\mathbf{X}_{k-1}^{(i)+}) \tag{23}$$
   b) Compute $\hat{\mathbf{x}}_k^-$ and P_k^-
$$\hat{\mathbf{x}}_k^- = \sum_{i=1}^{N} w^{(i)} \mathbf{X}_k^{(i)-} \tag{24a}$$
$$P_k^- = \sum_{i=1}^{N} w^{(i)} \left(\mathbf{X}_k^{(i)-} - \hat{\mathbf{x}}_k^-\right) \left(\mathbf{X}_k^{(i)-} - \hat{\mathbf{x}}_k^-\right)^T + Q_k \tag{24b}$$
   c) If $Q_k = 0$, $\mathbf{X}_k^{(i)-}$ need no transformation; if $Q_k \neq 0$, transform $\mathbf{X}_k^{(i)-}$ such that its mean and covariance are $\hat{\mathbf{x}}_k^-$ and P_k^-, respectively
3) Measurement Update
   a) Compute the quantities
$$\mathbf{Z}_k^{(i)-} = \mathbf{g}_k\left(\mathbf{X}_k^{(i)-}\right) \tag{25a}$$
$$\hat{\mathbf{z}}_k^- = \sum_{i=1}^{N} w^{(i)} \mathbf{Z}_k^{(i)-} \tag{25b}$$
$$P_k^{xz} = \sum_{i=1}^{N} w^{(i)} (\mathbf{X}_k^{(i)-} - \hat{\mathbf{x}}_k^-)(\mathbf{Z}_k^{(i)-} - \hat{\mathbf{z}}_k^-)^T \tag{25c}$$
$$P_k^{zz} = \sum_{i=1}^{N} w^{(i)} (\mathbf{Z}_k^{(i)-} - \hat{\mathbf{z}}_k^-)(\mathbf{Z}_k^{(i)-} - \hat{\mathbf{z}}_k^-)^T + R_k \tag{25d}$$
   b) Compute the Kalman gain K_k
$$K_k = P_k^{xz}(P_k^{zz} + R_k)^{-1} \tag{26}$$
   c) Compute $\hat{\mathbf{x}}_k^+$ and P_k^+
$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + K_k\left(\mathbf{z}_k - \hat{\mathbf{z}}_k^-\right) \tag{27a}$$
$$P_k^+ = P_k^- - K_k(P_k^{xz})^T \tag{27b}$$
   d) Transform $\mathbf{X}_k^{(i)-}$ to $\mathbf{X}_k^{(i)+}$ whose mean and covariance are $\hat{\mathbf{x}}_k^+$ and P_k^+, respectively

Note that the quadrature points are generated only once in the initialization. In the time update and measurement update, they are transformed but are not re-generated based on square root decomposition.

The existence of feasible linear transformations satisfying the mean and covariance constraints was established by construction in [11]. The mean is matched by a shift vector; the covariance is matched by a linear transformation matrix. Define the error matrices before and after transformation and the weights by

$$\tilde{X}^- = \begin{bmatrix}\mathbf{X}^{(1)-} - \hat{\mathbf{x}}^- & \cdots & \mathbf{X}^{(N)-} - \hat{\mathbf{x}}^-\end{bmatrix} \tag{28a}$$
$$\tilde{X}^+ = \begin{bmatrix}\mathbf{X}^{(1)+} - \hat{\mathbf{x}}^+ & \cdots & \mathbf{X}^{(N)+} - \hat{\mathbf{x}}^+\end{bmatrix} \tag{28b}$$
$$\mathbf{w} = \begin{bmatrix}w^{(1)} & \ldots & w^{(N)}\end{bmatrix}^T \tag{28c}$$
$$W = \mathrm{diag}(\mathbf{w}) \tag{28d}$$

The linear transformation is given by
$$\tilde{X}^+ = A\tilde{X}^- \text{or } \tilde{X}^+ = \tilde{X}^- B \tag{29}$$
with A and B matrices of appropriate dimensions. Note that for sparse grid quadrature filters, B is a larger matrix than A because the number of quadrature points is greater than the dimension of the state. The pre-multiplication transformation $\tilde{X}^+ = A\tilde{X}^-$ is use in this work.

The linear transformation from $\tilde{X}^-$ to $\tilde{X}^+$ should be such that if
$$\tilde{X}^-\mathbf{w} = \mathbf{0} \tag{30a}$$
$$\tilde{X}^- W(\tilde{X}^-)^T = P^- \tag{30b}$$
then $\tilde{X}^+$ satisfies the mean and covariance constraints, given by
$$\tilde{X}^+\mathbf{w} = \mathbf{0} \tag{31a}$$
$$\tilde{X}^+ W(\tilde{X}^+)^T = P^+ \tag{31b}$$

For $\tilde{X}^+ = A\tilde{X}^-$, the mean constraint is always satisfied because
$$\tilde{X}^+\mathbf{w} = A\tilde{X}^-\mathbf{w} = A\,\mathbf{0} = \mathbf{0} \tag{32}$$
The updated covariance matrix P^+ can be written as
$$\begin{aligned} P^+ &= (\tilde{X}^+)W(\tilde{X}^+)^T \\ &= (A\tilde{X}^-)W(A\tilde{X}^-)^T \\ &= A(\tilde{X}^-)W(\tilde{X}^-)^T A^T = AP^-A^T \end{aligned} \tag{33}$$
If P^+ and P^- are positive definite, they can both be expressed in terms of their square root factors, given by
$$P^+ = L^+(L^+)^T \tag{34a}$$
$$P^- = L^-(L^-)^T \tag{34b}$$
It follows that
$$P^+ = L^+(L^+)^T = AL^-(L^-)^T A^T \tag{35}$$
Hence, the transformation matrix A must satisfy
$$L^+ = AL^-Q \tag{36}$$
where Q is an arbitrary orthonormal matrix satisfying $QQ^T = I$, with I the identity matrix. Solving the equation gives
$$A = L^+Q^T(L^-)^{-1} \tag{37}$$
Setting Q to the identity matrix results in the simplest solution $A = L^+(L^-)^{-1}$. Because A is nonunique, $\tilde{X}^+$ is nonunique, either.

The optimal transformation is defined as the matrix that minimizes the difference between $\tilde{X}^+$ and $\tilde{X}^-$. Note that this is not necessarily optimal in the Bayesian sense. The idea is that the transformation should alter the distribution of the quadrature points as little as possible while matching the mean and covariance. Suppose the underlying pdf is asymmetric, the $\tilde{X}^+$ obtained this way is more likely to capture the higher moments of the pdf.

The optimal transformation is defined as the solution to the following minimization problem:

$$\begin{aligned} \min \quad & \left\| \tilde{X}^+ - \tilde{X}^- \right\|_F^2 \\ \text{subject to:} \quad & P^+ = AP^- A^T \end{aligned} \tag{38}$$

where $\|\cdot\|_F$ denotes the matrix Frobenius norm. The objective function may use weighed norms.

Generalization of the filter to non-additive noise cases is straightforward. The non-additive noise affects the way the means and covariances are computed, but the same algorithm is then used to transform the quadrature points.

IV. Example

The nonlinear filtering example of a vertically falling body is used in the test. It is a modified version of an example in [12]. The altitude, velocity, and ballistic coefficient of the falling body is estimated from the radar range measurements. The geometry is shown in Fig. 1, where the variables $x_1(t)$

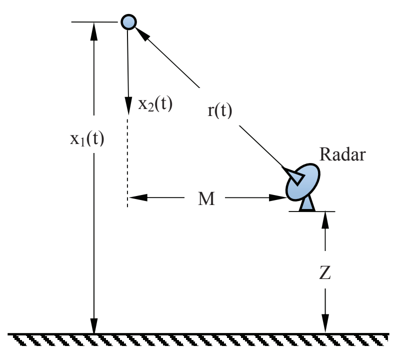


Fig. 1. Geometry of the Vertically Falling Body Example

and $x_2(t)$ denote the altitude and the downward velocity, respectively, $r(t)$ is the range, M is the horizontal distance, and Z is the radar altitude. The dynamic model is assumed to be

$$\dot{x}_1(t) = -x_2(t) \tag{39a}$$

$$\dot{x}_2(t) = -e^{-\alpha x_1(t)} x_2^2(t) x_3(t) + w(t) \tag{39b}$$

$$\dot{x}_3(t) = 0 \tag{39c}$$

where $w(t)$ is a zero-mean white noise process with power spectral density q, $x_3(t)$ is the ballistic coefficient and α is assumed to be a known constant. From the relative geometry of the radar and the falling body, the discrete-time range measurement at time k is given by

$$\tilde{y}_k = \sqrt{M^2 + (x_{1k} - Z)^2} + v_k \tag{40}$$

where the v_k is the zero-mean measurement noise with variance R. The values of the parameters are: $\alpha = 5 \times 10^{-5}$, $M = 1 \times 10^5$, $Z = 0$, $q = 10^{-8}$, and $R = 1 \times 10^4$. The measurement sampling period is set to 1.5 time units.

The true initial states are given by

$$x_1(0) = 3 \times 10^5, \quad x_2(0) = 2 \times 10^4, \quad x_3(0) = 1 \times 10^{-3}$$

The initial covariance for all filters is given by

$$P(0) = \begin{bmatrix} 1 \times 10^6 & 0 & 0 \\ 0 & 4 \times 10^6 & 0 \\ 0 & 0 & 1 \times 10^{-4} \end{bmatrix} \tag{41}$$

The true state time histories are obtained by numerically integrating the dynamic model using a fourth-order Runge-Kutta method with a step size of 1/64 time units. To account for the effect of the process noise, an effective discrete-time process noise vector is added to the propagated state vector after each propagation. The discrete-time process noise covariance is approximated by

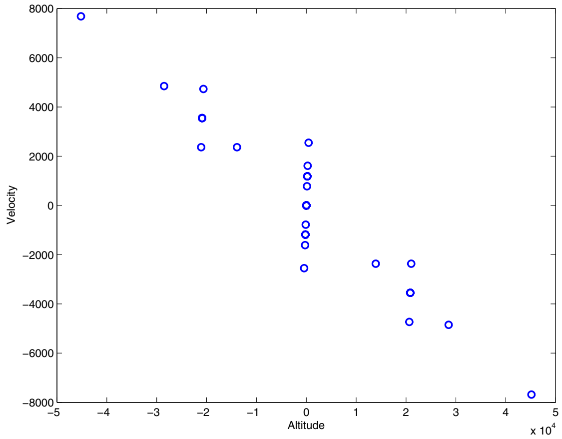
$$Q_k = \begin{bmatrix} 0 & 0 & 0 \\ 0 & q\Delta t & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

where Δt is the integration step size. The integration step sizes in the filters is 1.5 time units.

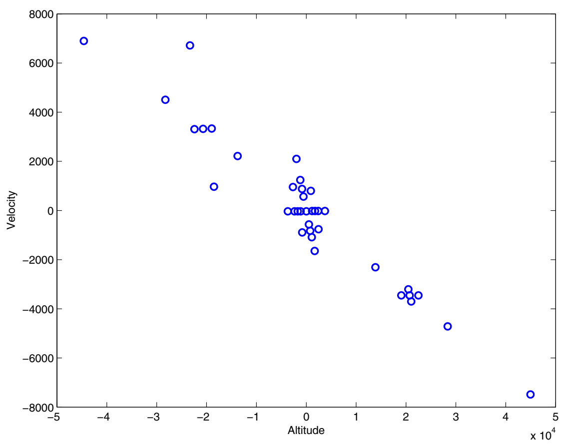
To illustrate the difference between the quadrature-based Gaussian filter and the extended sparse grid quadrature filter in how the points are updated, a 6-time-unit time update (with nonzero process noise) is run, in which the two methods use the same initial conditions. The results of the two methods are given by Fig. 2. Note that the two sets of points in Fig. 2(a) and Fig. 2(b) have the same weights, mean, and covariance, but different distributions and higher moments. For example, the symmetric points have zero odd moments, but the asymmetric points have nonzero odd moments. Because the effect of the process noise is small, the distribution of the points should resemble the one with zero process noise (Fig. 2(c)). Hence, the time update of the extended sparse grid quadrature filter results in better point distributions if the process noise is small.

Three extended sparse grid quadrature filters are compared with an unscented Kalman filter. The unscented Kalman filter uses the recommended value of the parameter $\kappa = 3 - n = 0$, which leads to zero weight on the central quadrature or sigma point. The effective number of the quadrature points is therefore $2n = 6$ instead of $2n+1 = 7$. Note that the unscented Kalman filter with this choice of κ coincides with the cubature Kalman filter [6]. The first two extended sparse grid quadrature filters use the same six quadrature points in the initialization. The sigma points of the unscented Kalman filter as well as the quadrature points of these two extended sparse grid quadrature filters have accuracy level 2, which means that they can be used to compute up to the third moments of a Gaussian random vector exactly. The third extended sparse grid quadrature filter uses a level-3 quadrature point set with 37 points. The level-3 set can be used to compute up to the fifth moments of a Gaussian random vector exactly.

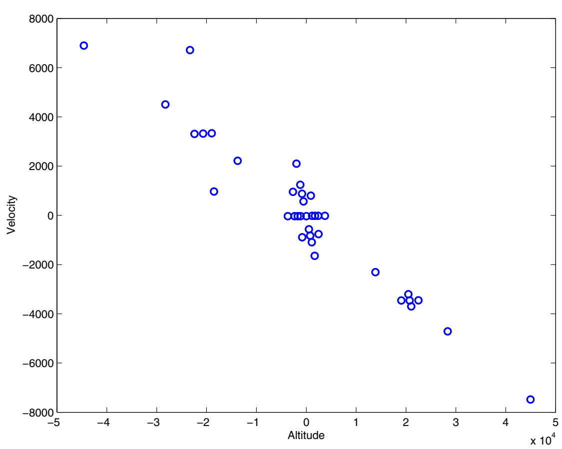
Of the three extended sparse grid quadrature filters, the first filter does not use the optimal transformation in the update but uses a feasible transformation that satisfies the covariance constraint and of the form $A = L^+(L^-)^{-1}$, where L^- and L^+ are given by Eqs. (34). Both the second and third direct extended sparse grid quadrature filters use the optimal transformation found by solving the optimization problems iteratively with the initial guess a feasible solution.



(a) Quadrature Based Gaussian Filter

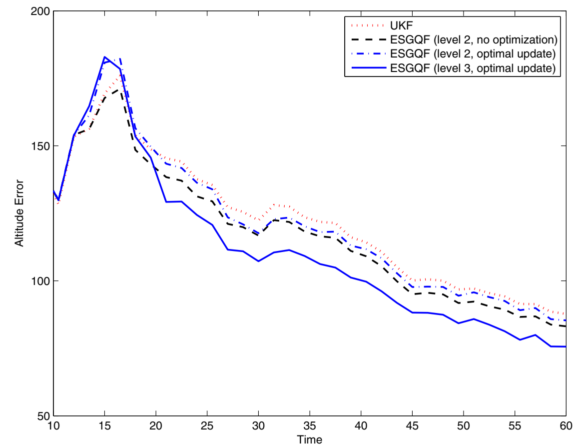


(b) Extended Sparse Grid Quadrature Filter

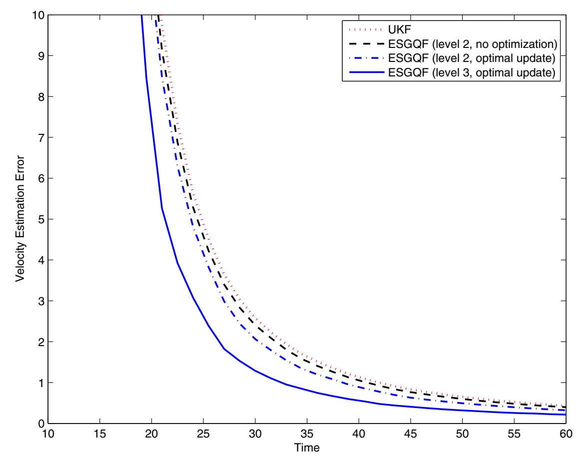


(c) Time Update Without Process Noise

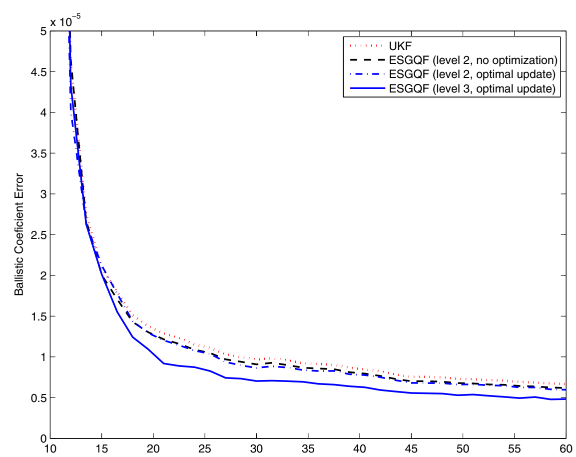
Fig. 2. Six-Second Time Update



(a) Altitude Error



(b) Velocity Error



(c) Ballistic Coeficient Error

Fig. 3. Roo-Mean-Square Errors over 50 Monte Carlo Runs

The root-mean-square estimation errors of the three extended sparse grid quadrature filters and the unscented Kalman filter over 50 runs are shown in Figs. 3, where UKF and ESGQF in the legends stand for the unscented Kalman filter and the extended sparse grid quadrature filter, respectively. The figures clearly show the performance differences between the filters. Specifically, the unscented Kalman filter is less accurate than the other filters and that the level-3 extended sparse grid quadrature filter is the most accurate. Comparing the three level-2 filters, we can see that the unscented Kalman filter, which does not transform the points directly, is less accurate than those that do.

V. Conclusions

By updating the quadrature points directly, the extended sparse grid quadrature filter avoids regenerating the quadrature points in each filter cycle and mitigates the dependence on the Gaussian approximation to non-Gaussian pdfs. The results of an illustrative example show that the extended sparse grid quadrature filter is more accurate than the unscented Kalman filter. The formulation of and numerical solution to the optimization problem of the extended sparse grid quadrature filter as well as the time complexity of the filter will be investigated in further work.

Aknowledgments

The work is partially supported by the National Science Foundation under Grant No. CMMI-1301111.

References

[1] K. Xiong and K. Ito, "Gaussian filters for nonlinear filtering problems," *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910–927, 2000.

[2] M. Šimandl and J. Duník, "Derivative-free estimation methods: New results and performance analysis," *Automatica*, vol. 45, no. 7, pp. 1749–1757, 2009.

[3] Y. Wu, D. Hu, M. Wu, and X. Hu, "A numerical-integration perspective on gaussian filters," *IEEE Transactions on Signal Processing*, vol. 54, no. 8, pp. 2910–2921, 2006.

[4] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new method for nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.

[5] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[6] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, 2009.

[7] ——, "Cubature Kalman smoothers," *Automatica*, vol. 47, no. 10, pp. 2245–2250, 2011.

[8] B. Jia, M. Xin, and Y. Cheng, "High-degree cubature Kalman filter," *Automatica*, vol. 49, no. 2, pp. 510–518, 2013.

[9] ——, "Sparse Gauss-Hermite quadrature filter with application to spacecraft attitude estimation," *Journal of Guidance, Control and Dynamics*, vol. 27, no. 2, pp. 367–379, 2011.

[10] ——, "Sparse-grid quadrature nonlinear filtering," *Automatica*, vol. 48, no. 2, pp. 327–341, 2012.

[11] Y. Tian and Y. Cheng, "Novel measurement update method for quadrature-based Gaussian filters," in *AIAA Guidance, Navigation, and Control Conference*, Boston, MA, August 19–22 2013.

[12] J. L. Crassidis and J. L. Junkins, *Optimal Estimation of Dynamic Systems*. Boca Raton, FL: CRC Press, 2011, ch. 3.